Fall 2008	**CPE 101:** Fundamentals of Computer Science I	Alexander Dekhtyar

Homework 3. . .

Due date: Monday, November 3

Submission Instructions

The assignment has only a programming part, which needs to be submitted before midnight on Monday. Due to the time-sensitive nature of parts of the assignment, there is no late submission policy

To submit, ssh to vogon and use the following `handin` command.

- Section 09:

```
> handin dekhtyar-grader hw03-09 <file1>,...,<fileN>
```

- Section 11:

```
> handin dekhtyar-grader hw03-11 <file1>,...,<fileN>
```

The list of files to submit is supplied below.

Assignment Preparation

Collaboration. This is an **individual assignment**. No collaboration is permitted.

Programming Style. All submitted C programs for the programming part must adhere to the programming style described in detail at

http://users.csc.calpoly.edu/∼cstaley/General/CStyle.htm

When graded, the programs will be checked for style. Any stylistic violations are subject to a 10% penalty.

Testing and Submissions. Any submission that does not compile using the

```
gcc -ansi -Wall -Werror -lm
```

compiler settings will receive an automatic score of 0.

Assignment.

Files to submit. You will be submitting three files:

`elections.c, elections.h, predictions.<UserName>`

First two files will contain your program. The third file will contain your *state-by-state* prediction of the November 4 US Presidential election. The collection of all predictions will serve as the set of test cases for all your programs.

The name of the prediction file is `prediction.<UserName>` where `<Username>` is your CSL login user name. For example my prediction test case is named `predictions.dekhtyar`.

To simplify generation of test cases, a template file `predictions.template` with all state names is available on the course web page.

Program: elections.c, elections.h

Change your `elections.c` program to take advantage of loops, functions and function libraries.

Input. The input to your program will be different. Results of elections in each state do not come in the alphabetical order - they typically come East-to-West. The new input format will allow your program to read the sequence of states in **any** order. The format is as follows:

- The first input parameter is a mode indicator. Value of 1 means verbose mode, Value of 0 means silent mode. All other values are invalid. The mode indicator is followed by a line break.
- Next, the input will consist of 51 lines. Each line will contain information about the election results in a single state. The format of each line is the same:

  ```
  ST x
  ```

  Here `S` and `T` are two **uppercase** characters, jointly representing the US Postal code for a specific state or Washington, DC (e.g., `AL, CA, WV, UT, RI`). `x` is a state result, which is a 0 if the state is won by John McCain and 1 if the sate is won by Barack Obama.

A sample input file is shown in Appendix A.

Modes. Your program shall operate in the same two modes, silent and verbose as the previous (Lab 3) version.

Structural Changes. The main changes in the program specifications are structural in nature. You shall create a `elections.h` header file. This file shall contain the following:

1. **All constant definitions**. Move them from the `elections.c` file.

2. `int getStateECVotes(char letter1, char letter2)` function definition.

   This function takes as input the two characters of the US Postal Service code for a state and returns back the number of Electoral College votes for this state. For example if the function call is:

   ```
    getStateECVotes('C','A')
   ```

   the function returns 55.

3. `void printStateMessage(char letter1, char letter2, int decision)` function definition.

   This function takes as input the two characters of the US Postal Service code for a state and an integer parameter `decision` specifying the decision of the state (0 - McCain, 1 - Obama). If Obama has won the state, the function prints the message

   ```
   <ST>  goes to Obama
   ```

   If McCain has won the state, the function prints the message

   ```
   <ST> goes to McCain
   ```

   Here, `<ST>` is the US Postal Service code for the state (also, the the values of `letter1` and `letter2` variables for the first and second letter of the code respectively).

   Note: There are two spaces between the state code and the word `"goes"` in the "Obama" message and only on space in the "McCain" message. This yields output that looks as follows:

   ```
   IA  goes to Obama
   KS goes to McCain
   ```

elections.c file. Your `elections.c` file now shall contain only the `main()` function. The `main()` function shall do the following:

1. Read in the mode indicator;
2. In a loop read the lines of the input.
3. For each line, update the EC scores of the candidates, print the state decision message if the mode is verbose.
4. Print the final results.

Printing the final results. Your program, in addition to tracking the number of electoral college votes each candidate received, shall also track the number of states each candidate won. This number shall be printed together with the EC vote total at the end of the program. The output line format now is:

```
Electoral College vote total for John  McCain (R) is xxx (zz states)
Electoral College vote total for Barack Obama (D) is yyy (ww states)
```

Here, `xxx` and `yyy` are electoral collect vote totals for McCain and Obama respectively, and `zz` and `ww` are the numbers of states (including DC, so they should add up to 51) McCain and Obama respectively won. See Appendix B for the full program output.

Appendix A: Sample input

Here is a sample input file.

```
> cat el-test01
1
AL 0
AK 0
AZ 0
AR 0
CA 1
CO 1
CT 1
DE 1
DC 1
FL 1
GA 0
HI 1
ID 0
IL 1
IN 1
IA 1
KS 0
KY 0
LA 0
ME 1
MD 1
MA 1
MI 1
MN 1
MS 0
MO 1
MT 0
NE 0
NV 1
NH 1
NJ 1
NM 1
NY 1
NC 1
ND 0
OH 1
OK 0
OR 1
PA 1
RI 1
SC 0
SD 0
TN 0
TX 0
UT 0
VT 1
VA 1
WA 1
WV 0
WI 1
WY 0
```

Appendix B: Sample run

Here is a sample run using the file from Appendix A as input.

```
> gcc -ansi -Wall -Werror -lm -o elections elections.c
> elections < el-test01
AL goes to McCain
AK goes to McCain
AZ goes to McCain
AR goes to McCain
CA  goes to Obama
CO  goes to Obama
CT  goes to Obama
DE  goes to Obama
DC  goes to Obama
FL  goes to Obama
GA goes to McCain
HI  goes to Obama
ID goes to McCain
IL  goes to Obama
IN  goes to Obama
IA  goes to Obama
KS goes to McCain
KY goes to McCain
LA goes to McCain
ME  goes to Obama
MD  goes to Obama
MA  goes to Obama
MI  goes to Obama
MN  goes to Obama
MS goes to McCain
MO  goes to Obama
MT goes to McCain
NE goes to McCain
NV  goes to Obama
NH  goes to Obama
NJ  goes to Obama
NM  goes to Obama
NY  goes to Obama
NC  goes to Obama
ND goes to McCain
OH  goes to Obama
OK goes to McCain
OR  goes to Obama
PA  goes to Obama
RI  goes to Obama
SC goes to McCain
SD goes to McCain
TN goes to McCain
TX goes to McCain
UT goes to McCain
VT  goes to Obama
VA  goes to Obama
WA  goes to Obama
WV goes to McCain
WI  goes to Obama
WY goes to McCain
Electoral College vote total for John  McCain (R) is 163 (21 states)
Electoral College vote total for Barack Obama (D) is 375 (30 states)
The next President of the United States is Barack Obama
```